



# Production Planning Using Mixed Lots

Enrico Canuto, Fabrice Chauvet, Jean-Marie Proth

## ► To cite this version:

Enrico Canuto, Fabrice Chauvet, Jean-Marie Proth. Production Planning Using Mixed Lots. [Research Report] RR-3113, INRIA. 1997, pp.19. inria-00073577

**HAL Id: inria-00073577**

**<https://inria.hal.science/inria-00073577>**

Submitted on 24 May 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *Production Planning Using Mixed Lots*

Enrico Canuto - Fabrice Chauvet  
Jean-Marie Proth

N° 3113  
Février 1997

THÈME 4



*apport  
de recherche*

Les rapports de recherche de l'INRIA  
sont disponibles en format postscript sous  
ftp.inria.fr (192.93.2.54)

si vous n'avez pas d'accès ftp  
la forme papier peut être commandée par mail :  
e-mail : dif.gesdif@inria.fr  
(n'oubliez pas de mentionner votre adresse postale).

par courrier :  
Centre de Diffusion  
INRIA  
BP 105 - 78153 Le Chesnay Cedex (FRANCE)

INRIA research reports  
are available in postscript format  
ftp.inria.fr (192.93.2.54)

if you haven't access by ftp  
we recommend ordering them by e-mail :  
e-mail : dif.gesdif@inria.fr  
(don't forget to mention your postal address).

by mail :  
Centre de Diffusion  
INRIA  
BP 105 - 78153 Le Chesnay Cedex (FRANCE)

# PRODUCTION PLANNING USING MIXED LOTS

Enrico CANUTO\*

Fabrice CHAUVET\*\*

Jean-Marie PROTH\*\*\*

## ABSTRACT:

In this paper, we consider the production planning of a manufacturing system capable of producing several types of part. The demands are known at the end of each elementary period (at the end of each day or week for instance). The objective function (i.e. the criterion) is the sum of inventory, backloging and set-up costs on a finite horizon. We use the concept of lot to simplify management decisions: instead of launching in production parts type by type, we launch lots. A lot is a mix of parts of different types. The drawback is the definition of the manufacturing times of the lots. We consider that the manufacturing time of a lot is its manufacturing time in steady state: it is a way to associate a manufacturing time to each lot, independently from the work-in-process and the set-up times. As a consequence, we have to launch identical lots in series to reduce the error made by introducing this hypothesis. In this paper, we propose a heuristic algorithm and a branch-and-bound approach to solve this problem. Numerical examples are provided.

## KEYWORDS:

Branch-and-bound, Esprit project HIMAC, Linear programming, Lots, Production planning.

\* Politecnico di Torino, Dipartimento di Automatica e Informatica, Corso Duca degli Abruzzi, 24, 10129 Torino, ITALY.

\*\* INRIA-Lorraine, Technopôle Metz 2000, 4 rue Marconi, 57070 Metz, FRANCE.

\*\*\* INRIA-Lorraine, Technopôle Metz 2000, 4 rue Marconi, 57070 Metz, FRANCE, and Institute for Systems Research, University of Maryland, College Park 20742, USA.

*This work has been done under the umbrella of the ESPRIT HIMAC Project supported by the European Communities.*

# PLANIFICATION DE LA PRODUCTION UTILISANT DES LOTS COMPOSES DE PRODUITS DIFFERENTS

Enrico CANUTO\*

Fabrice CHAUVET\*\*

Jean-Marie PROTH\*\*\*

## RESUME :

Dans ce papier, nous nous intéressons à la planification d'un système de production qui fabrique différents types de produits. Les demandes sont connues à la fin de chaque période élémentaire (le jour ou la semaine, par exemple). Le critère à optimiser est la somme des coûts de stockage, de rupture et de préparation sur un horizon fini. Nous utilisons la notion de lot pour simplifier les décisions de gestion : au lieu de lancer en production des commandes type de produits par type de produits, nous lançons des lots. Un lot est un ensemble de produits de différents types. Le problème réside dans la définition du temps de fabrication d'un lot. Nous considérons que le temps de fabrication d'un lot est son temps de fabrication en régime permanent : c'est un moyen d'associer un temps de fabrication à chaque lot, indépendamment des en-cours et des temps de préparation. En conséquence, nous devons lancer des séquences de lots identiques suffisamment longues afin d'atténuer l'erreur systématique introduite par cette hypothèse. Dans ce papier, nous proposons une heuristique et une approche par séparation et évaluation pour résoudre le problème. Des exemples numériques sont proposés.

## MOTS-CLEFS :

Séparation et évaluation, Projet ESPRIT HIMAC, Programmation linéaire, Lots, Planification de la production.

\* Politecnico di Torino, Dipartimento di Automatica e Informatica, Corso Duca degli Abruzzi, 24, 10129 Torino, ITALY.

\*\* INRIA-Lorraine, Technopôle Metz 2000, 4 rue Marconi, 57070 Metz, FRANCE.

\*\*\* INRIA-Lorraine, Technopôle Metz 2000, 4 rue Marconi, 57070 Metz, FRANCE, et Institute for Systems Research, University of Maryland, College Park 20742, USA.

*Ce travail a été réalisé dans le cadre du projet ESPRIT HIMAC co-financé par les Communautés Européennes.*

## 1. INTRODUCTION

Production planning activity usually involves a quite large horizon, which makes it difficult to consider the detailed entities (part types, machines). To avoid computational burden, a common way is to aggregate entities and to make decisions on sets of parts (lots) and sets of machine (subsystems).

The purpose of this paper is to develop an adequate model for planning the production of a subsystem, assuming that the demands are known at the end of each elementary period (i.e. at the end of each day or week, depending on the type of production). The basic element of the model proposed hereafter is the lot. A lot is a mix of part types. Different types of lot are available. Each type of lot is characterised by a mix of parts. The set of lot types covers a large variety of mixes in order to fit "at the best" the demand. The changeover of the production from one type of lot to another introduces a set-up time and a set-up cost. The production planning consists in defining the sequence of lots to be manufactured in order to minimise, on a given horizon, the sum of inventory, backloging, and set-up costs.

The main difficulty of this problem consists in assigning a manufacturing time to each lot. We decided, according to the results obtained in the ESPRIT project HIMAC (see [2] and [3]), to consider that the manufacturing time of a lot is its manufacturing time in steady state: it is a way to keep the manufacturing times of the lots independent from the work-in-process and set-up times. To attenuate the consequences of this systematic error, we introduce a parameter  $T_m$  which defines the minimal period on which identical lots should be produced. Parameter  $T_m$  is the control parameter used to disconnect the planning level from the next low level which is the scheduling level.

Note that another way to disconnect the planning level from the scheduling level is to over-evaluate the manufacturing times of parts, and thus to consider that the manufacturing time associated to a part is also the manufacturing time of any set of parts of the same type, assuming that the size of this set remains limited. It is the way used in the MRP approaches (see [5] and [9]).

Note also that the existing approaches using lots (see for instance [4], [7], [11] and [12]) do not take into account the capacity of the manufacturing system (i.e. the manufacturing times) but only the sizes of lots to be manufactured in sequence and which are composed of identical parts. Their goal is mainly to make a trade-off between inventory costs and set-up costs. The model presented in this paper can be considered as a generalisation of these models.

The approach presented in this paper concerns the short-term planning level of a hierarchical production management system. The main advantage of this approach is that the decisions to be made at the next low level, that is the scheduling decisions, can be made off-line for each lot, based on the average work-in-process. Another important remark is that the same approach can apply to all the planning levels, taking into account the following rule: the higher the level, the more aggregated the entities, i.e. the more aggregated the resources and the bigger the lots.

The problem is formulated in Section 2. Section 3 presents a heuristic approach and a branch-and-bound approach to solve the problem. Section 4 is the conclusion.

In the remaining of this paper, we use bold letters for vectors. All the vectors are row vectors. Column vectors are obtained by transposing row vectors.  $t$  denotes the transpose operation.

## 2. PROBLEM FORMULATION

A manufacturing system (MS) is composed of  $m$  machines  $M_1, M_2, \dots, M_m$ . This system can manufacture  $n$  types of parts denoted by  $P_1, \dots, P_n$ . A part type is defined by its manufacturing process, which is the sequence of machines to be visited in order to complete one unit of this type of part, and by the time one unit of part type spends on each machine. The manufacturing process associated with part type  $P_i$  is, for instance:

$$P_i: M_1^i(\theta_1^i), M_2^i(\theta_2^i), \dots, M_{q_i}^i(\theta_{q_i}^i)$$

where  $M_k^i \in \{M_1, \dots, M_m\}$  for  $i = 1, \dots, n$  and  $k = 1, \dots, q_i$ .

The time axis is divided into elementary periods, the length of which is denoted by  $\Delta$ . For instance, if the customers' requirements are expressed each and every day, then the length of  $\Delta$  is a day.  $\mathbf{D}_r$  is the row vector the  $i$ -th component of which is  $d_r^i$  for  $i = 1, \dots, n$ .

We denote by  $d_r^i$ ,  $i = 1, \dots, n$ ,  $r = 1, 2, \dots$ , the customers' requirements (i.e. the demands) of product  $P_i$  at the end of the  $r$ -th elementary period (i.e. at time  $r \Delta$ ).

We use the concept of lot. A lot is a mix of parts of different types which have to be launched in the system simultaneously and manufactured during the same period of time. Assuming that  $J$  lots are in use, lot  $\mathbf{L}^j$ ,  $j = 1, \dots, J$ , is defined as:

$$\mathbf{L}^j = [l_1^j, \dots, l_n^j]$$

where  $l_i^j$  is the number of parts of type  $P_i$  in a lot  $\mathbf{L}^j$ .

Let us assume that a given lot  $\mathbf{L}^j$  is manufactured in steady state. A steady state is reached when, producing a sequence of lots  $\mathbf{L}^j$ , we observe that the manufacturing time of a lot becomes constant. This time is the time needed by the bottleneck machine to absorb one lot, assuming that the required semi-finished products are available in front of this machine. It is computed as follows:

$$\bar{\theta}^j = \max_{s \in \{1, \dots, m\}} \sum_{(k,i) \in E_s} l_i^j \cdot \theta_k^i \quad (1)$$

where  $E_s = \{(k,i) / i = 1, \dots, n, k = 1, \dots, q_i, M_k^i = M_s\}$

### Example

Let us assume that MS is composed of three machines denoted by  $M_1, M_2$  and  $M_3$ . Assume also that

MS can manufacture three product types, say  $P_1$ ,  $P_2$  and  $P_3$ , whose manufacturing processes are given hereafter (the figures in parentheses are the manufacturing times):

$P_1$ :  $M_1(5)$ ,  $M_2(2)$ ,  $M_3(4)$

$P_2$ :  $M_2(6)$ ,  $M_1(1)$

$P_3$ :  $M_1(1)$ ,  $M_3(4)$

Let us consider the lot  $L^1 = \{3, 1, 1\}$ . The related manufacturing times are obtained as follows:

Time needed by  $M_1$  to perform one lot  $L^1$ :  $3 \times 5 + 1 \times 1 + 1 \times 1 = 17$

Time needed by  $M_2$  to perform one lot  $L^1$ :  $3 \times 2 + 1 \times 6 = 12$

Time needed by  $M_3$  to perform one lot  $L^1$ :  $3 \times 4 + 1 \times 4 = 16$

Thus, the bottleneck machine is  $M_1$ , and  $\bar{\theta}^1 = 17$ .

Let us now consider lot  $L^2 = \{1, 5, 5\}$ . In this case:

Time needed by  $M_1$  to perform one lot  $L^2$ :  $1 \times 5 + 5 \times 1 + 5 \times 1 = 15$

Time needed by  $M_2$  to perform one lot  $L^2$ :  $1 \times 2 + 5 \times 6 = 32$

Time needed by  $M_3$  to perform one lot  $L^2$ :  $1 \times 4 + 5 \times 4 = 24$

In this case, the bottleneck machine is  $M_2$ , and  $\bar{\theta}^2 = 32$ .

Optimising the management of such a system consists in finding a schedule which meets the demands at the best, i.e. which minimises the total cost obtained by summing up the inventory costs, the backlogging costs and the set-up costs related to the schedule. We denote by:

$I_i$  the cost of keeping in stock one part of type  $P_i$  during one elementary period  $\Delta$ ,

$B_i$  the cost incurred when one part of type  $P_i$  is missing during one elementary period  $\Delta$ .

In addition to the inventory and backlogging costs, we consider:

(i) The set-up time. We denote by  $\mu(j_1, j_2)$  the set-up time required when switching from lot  $L^{j_1}$  to lot  $L^{j_2}$ . In other words,  $\mu(j_1, j_2)$  is the time required for tool adjustment and preparation when we want to manufacture a lot  $L^{j_2}$  after a lot  $L^{j_1}$ .  $\beta(j_1, j_2)$  is the cost corresponding to  $\mu(j_1, j_2)$ .

(ii) A weight  $w$ , which is a control parameter.

Such a scheduling problem is known to be NP-hard. It is the reason why we introduced lots. Using lots, the scheduling problem is re-written as follows:

**Define the sequence of lots which meets "at the best" (i.e. while minimising the total cost) customers' requirements.**

This new formulation is a simplified form of the previous one. It assumes that an optimal schedule has been previously computed off-line for each lot. It also assumes that the manufacturing times of the lots are known and, since we know these manufacturing times only in steady state (see relation (1)), it turns out that the solution we are looking for is the concatenation of sequences of identical lots, called



elementary sequences, each sequence being long enough to not only reach the steady state, but also to reduce as much as possible the error due to the transient states which occur when switching from a sequence of identical lots to another sequence made of different identical lots.

In the following, we consider that the cost to be paid when switching from  $L^{j_1}$  to  $L^{j_2}$  is  $w\beta(j_1, j_2)$ . As a consequence, the greater  $w$ , the smaller the number of switchings from one lot to another one in the optimal solution. Indeed, this is obtained at the expense of backlogging and inventory costs.

The goal of this paper is to find the best sequence of elementary sequences (i.e. of sequences of identical lots). We propose two algorithms to solve this problem. The first one is a heuristic algorithm and the second one, which provides an optimal solution, is a branch-and-bound approach (see [6] and [8]).

### 3. LOT SCHEDULING

#### 3.1. Formulation of the scheduling problem

When the level of the demands is low, we have to keep the system idle during some periods. It is the reason why we introduce the so-called "empty lot" denoted by  $L^0$ . Such a lot is composed with zero unit of each type of product, but requires a time  $\bar{\theta}^0$  to be performed. We will choose the value of  $\bar{\theta}^0$  in the next subsection. The set-up time and the set-up cost  $\mu(q, 0)$  and  $\beta(q, 0)$  between a lot  $L^q$  and a lot  $L^0$  are equal to zero for any  $q \in \{1, \dots, Q\}$ . The set-up time and the set-up cost  $\mu(0, q)$  and  $\beta(0, q)$  between a lot  $L^0$  and a lot  $L^q$  are equal to  $\mu(q^*, q)$  and  $\beta(q^*, q)$  respectively, where  $L^{q^*}$  is the last lot different from  $L^0$  which precedes  $L^q$  in the sequence under consideration, i.e.  $\langle \dots L^{q^*}, L^0, L^0, \dots, L^0, L^q \dots \rangle$ .

In the following, we assume that  $\bar{\theta}^q$  is the manufacturing time of a lot  $L^q$ , whatever  $q \in \{0, \dots, Q\}$ . The goal of the scheduling process is to reach a result which exposes large sequences of identical lots, so as to reduce the manufacturing times error, but which meets the demand at the best.

Let  $H$  be the horizon at which the demand is known. In other words,  $D_r$  is known for  $r = 1, \dots, H$ . We are looking for a sequence of lots  $\{L^{i_1} L^{i_2} \dots L^{i_z}\} = \tilde{L}$  such that:

$$\sum_{s=1}^{z-1} \bar{\theta}^{i_s} + \sum_{s=0}^{z-2} \mu(i_s, i_{s+1}) < H\Delta \leq \sum_{s=1}^z \bar{\theta}^{i_s} + \sum_{s=0}^{z-1} \mu(i_s, i_{s+1}) \quad (2)$$

Relation (2) reflects the fact that the manufacturing time of the sequence, including the set-up times, covers horizon  $H$ , i.e. period  $[0, H\Delta]$ , at the best.  $L^{i_0}$ ,  $i_0 \in \{1, \dots, q\}$ , is the last lot completed by the starting time of the computation (i.e. by time 0).

Let  $v_\tau$  be the maximal number of lots of sequence  $\tilde{L}$  which can be completed until time  $\tau$ ,  $\tau \geq 0$ .

$\bar{\mathbf{D}}^\tau = \sum_{r/\tau\Delta \leq \tau} \mathbf{D}_r$  is the sum of the demand vectors until time  $\tau$  (including  $\tau$ ) and  $\bar{\mathbf{U}}^\tau = \sum_{s=1}^{\nu_\tau} \mathbf{L}^{i_s}$  is the

production vector until time  $\tau$ . Note that only the lots which are completed at time  $\tau$  are taken into account.

We denote by  $\mathbf{I}$  the vector whose components are the  $I_i$ ,  $i = 1, \dots, n$ , defined before, i.e. the inventory costs of the various types of part, and by  $\mathbf{B}$  the vector whose components are the  $B_i$ ,  $i = 1, \dots, n$ , i.e. the backlogging costs of the various types of part.

Furthermore,  $\mathbf{S}^0$  is the vector whose components are the initial inventories of the different types of components, i.e. the inventories at time 0.

We denote by  $\tau_1, \dots, \tau_u$  the points in time when either demands occur or a lot is completed, and we assume that  $0 = \tau_0 < \tau_1 < \dots < \tau_u$ , where  $\tau_u$  is the time when the last lot is completed, according to (2).

If  $\mathbf{V}$  is a vector,  $\mathbf{V}^+$  is derived from  $\mathbf{V}$  by replacing all the negative components by 0.

Taking into account the above definitions, the cost corresponding to sequence  $\tilde{\mathbf{L}}$  is:

$$C(\mathbf{S}^0, \tilde{\mathbf{L}}) = \sum_{k=0}^{u^*-1} \left[ \mathbf{S}^0 + \bar{\mathbf{U}}^{\tau_k} - \bar{\mathbf{D}}^{\tau_k} \right]^+ \mathbf{I}^t (\tau_{k+1} - \tau_k) + \sum_{j=0}^{H-1} \left[ \bar{\mathbf{D}}^{j\Delta} - \bar{\mathbf{U}}^{j\Delta} - \mathbf{S}^0 \right]^+ \mathbf{B}^t \Delta + w \sum_{s=0}^{z-1} \beta(i_s, i_{s+1}) \quad (3)$$

with  $\bar{\mathbf{D}}^{\tau_0} = \bar{\mathbf{U}}^{\tau_0} = 0$ .  $u^*$  is the greatest integer such that  $\tau_{u^*} < H\Delta$ . This choice was made to stop the computation of the inventory cost at horizon  $H$ .

In this equality:

- (i) The first sum of the right-hand side is the inventory cost. Since the inventory level may change each time a lot is completed or a demand occurs, each term corresponds to a period  $[\tau_k, \tau_{k+1}]$ ,  $k = 0, \dots, u^*-1$ .
- (ii) The second sum of the right-hand side is the backlogging cost. A backlog which arises at the beginning of an elementary period remains the same until the end of this period since deliveries occur at the end of each elementary period.
- (iii) The third sum of the right-hand side is the weighted set-up cost ( $w$  is the weight).

Finally, the problem to be solved can be written:

**Find  $\tilde{\mathbf{L}}$  which minimises (3) while satisfying constraint (2).**

Indeed, the lots belonging to  $\tilde{\mathbf{L}}$  are chosen among the selected lots.

This system is supposed to work on a rolling horizon basis.

### 3.2. Property of solutions

In this subsection, we present a property of importance which concerns the idle periods and leads to the choice of  $\bar{\theta}^0$ . The following definition is required to establish this property.

#### Definition

Let  $\delta$  be the largest time interval such that:

$\Delta = k\delta$ ,  $k$  being a strictly positive integer,

$\bar{\theta}^q = k_q\delta$  for  $q = 1, \dots, Q$ , the  $k_q$  values being strictly positive integer numbers,

$\mu(r,s) = k_{r,s}\delta$  for any pair  $(r,s) \in \{1, \dots, Q\} \times \{1, \dots, Q\}$ ; the  $k_{r,s}$  values being positive integer numbers.

Then  $\delta$  is called the Largest Divisor Period, or LDP for short. Note that  $\delta$  exists if we assume that  $\Delta$ , the manufacturing times and the set-up times are rational.

The following result holds.

#### Result 1

Let  $L^1, \dots, L^Z$ ,  $i_j \in \{1, \dots, Q\}$  for  $j = 1, \dots, Z$ , be a sequence of lots manufactured in the system considered, and  $E_1, \dots, E_R$  be the idle periods put in between some pairs of consecutive manufacturing periods; we denote this schedule by  $G_0$ . Then there exists a schedule  $G^*$  in which:

- (i) the manufacturing order of the lots is the same as in  $G_0$ ,
- (ii) the idle periods are multiples of the LDP  $\delta$ ,
- (iii) the cost related to  $G^*$  is less than or equal to the cost related to  $G_0$ .

#### Proof

Let  $t_r^b$  (resp.  $t_r^e$ ) be the beginning time (resp. the ending time) of the idle period  $E_r$ ,  $r = 1, \dots, R$ . We denote by  $t_{R+1}^b$  the completion time of  $L^Z$  and we set  $t_{R+1}^e = +\infty$ . Let  $r^*$  be the lowest integer  $r$  such that the duration of  $E_{r^*}$ , say  $d(E_{r^*})$ , is not a multiple of  $\delta$ , i.e.  $d(E_{r^*}) = \omega_{r^*}\delta$ ,  $\omega_{r^*}$  being a strictly positive and non integer number.

We set:

$$A_r = \lceil \omega_r \delta \rceil - \omega_r \delta$$

where  $\lceil \bullet \rceil$  denotes the lowest integer greater than or equal to  $\bullet$ .

Let  $k^*$  be the lowest integer such that:

$$\sum_{i=1}^{k^*} d(E_{r^*+i}) \geq A_{r^*}$$

We change the schedule  $G_0$  into  $G_1$  by:

- translating the block starting at time  $t_{r^*}^e$  and ending at time  $t_{r^*+1}^b$  of  $A_{r^*}$  to the right,
- if  $k^* \geq 1$ , translating the block starting at time  $t_{r^*+i}^e$  and ending at time  $t_{r^*+i+1}^b$  of  $A_{r^*} - \sum_{j=1}^i d(E_{r^*+j})$  to the right for  $i = 1, \dots, k^*-1$ .

With these translations, which are all less than  $\delta$ , the completion times of all the lots are either increased or remain the same. Furthermore, these completion times remain in the same elementary period as they were before the translation. As a consequence and according to relation (3), only the inventory cost may change, and if it changes, it decreases since the duration of keeping in stock some parts decreases.

At this point, the starting time of the first idle period such that  $d(E_r)$  is not a multiple of  $\delta$  increases.

We restart the same process until all the idle periods such that  $d(E_r)$  is not a multiple of  $\delta$  disappear, and we obtain  $G^*$  which verifies the conditions of Result 1. Q.E.D.

A consequence of Result 1 is that we set  $\bar{\theta}^0 = \delta$ , since we know that an optimal solution in which the durations of the idle periods are multiple of  $\delta$  always exist.

### 3.3. A heuristic algorithm

The heuristic algorithm presented hereafter consists in building a sequence of elementary sequences, each elementary sequence being composed of identical lots.

If  $\tilde{L}^s$  is the  $s$ -th elementary sequence obtained using the heuristic algorithm, then the final sequence, denoted by  $\tilde{L}$ , can be written as follows:

$$\tilde{L} = \tilde{L}^1 \circ \tilde{L}^2 \circ \dots \circ \tilde{L}^z$$

where  $\circ$  denotes the concatenation and  $z$  is the number of elementary sequences.

The heuristic algorithm depends on the integer parameter  $T_m$  defined in the introduction. Note that  $T_m$  does not hold for empty lots, since an idle period can be limited to  $\delta$ . Let us assume that  $\tilde{L}^s$  is composed of lots of type  $L^{f(s)}$ . Assume also that the elementary sequence  $\tilde{L}^{s-1}$  is completed at time  $\lambda_{s-1}$ . We set  $\tau_0 = \lambda_0 = 0$ , and we assume that the last elementary sequence completed before 0 is composed of lots of

type  $L^{f(0)}$ .  $\lambda_{s-1}$  is one of the points in time  $\tau_i$  defined in Section 4.1. Indeed,  $\lambda_{s-1} \geq \tau_{s-1}$ , since the set  $\{\tau_0, \tau_1, \dots, \tau_u\}$  also includes the points in time when a demand occurs.

The goal of the heuristic algorithm is to provide a sequence  $\tilde{L}$  such that  $C(S^0, \tilde{L})$  is as low as possible. We restrict the computation of  $\tilde{L}$  to time  $H\Delta - 2T_m$  since the heuristic algorithm presented hereafter uses a look-ahead period of length  $2T_m$ . This is not restrictive since the heuristic algorithm will work on a rolling horizon basis. The cost used in the algorithm is slightly different from the cost provided by relation (3). It is given by relation (4):

$$C(S^0, \tilde{L}) = \sum_{k=0}^{u^*-1} [S^0 + \bar{U}^{\tau_k} - \bar{D}^{\tau_k}]^+ I^t(\tau_{k+1} - \tau_k) + \sum_{k=0}^{u^*-1} [\bar{D}^{\tau_k} - \bar{U}^{\tau_k} - S^0]^+ B^t(\tau_{k+1} - \tau_k) + w \sum_{s=0}^{z-1} \beta(i_s, i_{s+1}) \quad (4)$$

In this formulation, the lots are taken into account as soon as they are completed, even in case of backlog: this is the difference with formulation (3). This allows to evaluate the future consequence of an event, i.e. the evolution of the backlogging at the end of the current elementary period.

The heuristic algorithm is presented hereafter. It consists in adding, at each iteration, two elementary sequences to a partial sequence. The elementary sequences selected are the ones:

- (i) the lengths of which are constrained by  $T_m$ , i.e. each elementary sequence covers at least one period  $T_m$ ,
- (ii) which leads to the minimal average value per time unit of the criterion for the whole sequence when adding the new partial sequences.

Only the first elementary sequence is added to the partial sequence, and the next iteration starts with this new partial sequence. The algorithm is presented hereafter.

### Algorithm HSCHEDULE

1. The following set of information is known:

- $\tau_0 = 0$ , starting time of the computation,
- $L^{f(0)}$ , last type of lots manufactured before time  $\tau_0 = \lambda_0$ ,
- $T_m$  parameter which obliges each set of identical lots to cover a minimal period,
- the set of lots available,
- the inventory and backlogging costs,
- the horizon  $H$ ,
- the elementary period  $\Delta$ ,
- the set-up times  $\mu(\bullet, \bullet)$ ,

- the set-up costs  $\beta(\bullet, \bullet)$ ,
- the weight  $w$ ,
- the demands at the end of each elementary period,
- the manufacturing times of the lots.

2. Set  $\tilde{L}(0) = \emptyset$  and  $\lambda_0 = 0$

3. For  $s = 1, 2, 3, \dots$ , do

3.1. Set  $C^* = +\infty$ ,  $r^* = 0$

3.2. Compute

$$a = \text{Min}(H\Delta - 2T_m - \lambda_{s-1}, T_m\Delta)$$

$$b = H\Delta - 2T_m - \lambda_{s-1}$$

3.3. For  $q = 0$  to  $Q$ , do

3.3.1. Compute

$$n_m(q) = \left\lceil [a - \mu(f(s-1), q)] / \bar{\theta}^q \right\rceil$$

$$n_M(q) = \left\lceil [b - \mu(f(s-1), q)] / \bar{\theta}^q \right\rceil$$

where  $\lceil \bullet \rceil$  is the lowest integer greater than or equal to  $\bullet$

If  $q = 0$ ; set  $n_m(q) = 1$

*This reflects the fact that an elementary sequence composed of empty lots may contain only one lot.  $n_m(q)$  and  $n_M(q)$  are respectively the minimal and the maximal number of lots of type  $L^q$  which can be introduced in the next elementary sequence.*

3.3.2. For  $r = \text{Max}(n_m(q), 1)$  to  $\text{Max}(n_M(q), 1)$ , do

3.3.2.1. Compute

$$v = \left\lceil [\lambda_{s-1} + \mu(f(s-1), q)] + r\bar{\theta}^q \right\rceil$$

*$v$  is the time at which  $r$  lots of type  $L^q$  are completed*

3.3.2.2. Compute

$$a_1 = \text{Min}(H\Delta - v, T_m\Delta)$$

$$b_1 = H\Delta - v$$

3.3.2.3. For  $q_1 = 0$  to  $Q$ , do

3.3.2.3.1. Compute

$$n_{1,m}(q_1) = \left\lceil [a_1 - \mu(q, q_1)] / \bar{\theta}^{q_1} \right\rceil$$

$$n_{1,M}(q_1) = \left\lceil [b_1 - \mu(q, q_1)] / \bar{\theta}^{q_1} \right\rceil$$

3.3.2.3.2. For  $r_1 = \text{Max}(n_{1,m}(q_1), 1)$  to  $\text{Max}(n_{1,M}(q_1), 1)$ , do

3.3.2.3.2.1. Compute

$$v_1 = \lceil v + \mu(q, q_1) + r_1 \bar{\theta}^{q_1} \rceil$$

$v_1$  is the time at which  $r_1$  lots of type  $L^{q_1}$  are completed in series with  $r$  lots of type  $L^q$  after  $\tilde{L}(s-1)$ .

3.3.2.3.2.2. Compute

$$y = C(S^0, \tilde{L}(r, q, r_1, q_1, s-1))$$

where  $\tilde{L}(r, q, r_1, q_1, s-1)$  is the concatenation of  $\tilde{L}(s-1)$ , the elementary sequence  $\tilde{L}(r, q)$  made with  $r$  lots of type  $L^q$ , and the elementary sequence  $\tilde{L}(r_1, q_1)$  made with  $r_1$  lots of type  $L^{q_1}$ . Cost  $y$  is computed using relation (4).

3.3.2.3.2.3. If  $y/v_1 < C^*$ , do

$$C^* = y/v_1, \quad r^* = r, \quad q^* = q, \quad v^* = v$$

3.3.2.3.2. End of loop  $r_1$

3.3.2.4. End of loop  $q_1$

3.3.3. End of loop  $r$

3.4. End of loop  $Q$

3.5. Set  $\lambda_s = v^*$

3.6. Set  $\tilde{L}(s) = \tilde{L}(s-1) \circ \tilde{L}(r^*, q^*)$

3.7. Set  $f(s) = q^*$

3.8. If  $v^* > H\Delta - 2T_m$ , stop.  $\tilde{L}(s)$  is the solution and  $CC = C^*v^*$  is the corresponding cost.

4. End of loop  $s$

### Remark

$T_m$  is the minimal length of the period covered by an elementary sequence. The goal of  $T_m$  is to guaranty that the manufacturing times  $\bar{\theta}^q$  associated with the type  $L^q$  of lots leads to an error which is as limited as possible; this remark is limited to  $q > 0$ , i.e. it does not concern empty lots. Note that the last sequence may cover a period which is shorter than  $T_m$ , but the consequences of this fact are negligible since the method presented in this paper should be associated with a rolling horizon process.

### A numerical example

We consider a manufacturing system which manufactures two types of products denoted by  $P_1$  and  $P_2$ . The demands at the end of the twenty five next elementary periods are given in Table 1.

Table 1: Demands

EP	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
$P_1$	5	3	4	4	9	8	0	5	4	22	8	7	3	4	9	8	5	6	24	14	26	18	9	7	8
$P_2$	9	7	10	22	26	10	10	20	5	8	7	5	10	7	5	9	6	5	10	4	3	2	5	4	3

EP = Elementary periods

The lots used in this case are:

$$L^1[2,26], L^2[6,10], L^3[9,9], L^4[7,4], L^5[22,1]$$

The length of one elementary period is 1 unit of time, and:

$$\bar{\theta}^1 = 1, \bar{\theta}^2 = 0.6, \bar{\theta}^3 = 0.6, \bar{\theta}^4 = 0.4, \bar{\theta}^5 = 0.8$$

The set-up times are given in Table 2. As a consequence,  $\bar{\theta}^0 = 0.2$ .

Table 2: Set-up times

$\wedge$	$L^1$	$L^2$	$L^3$	$L^4$	$L^5$
$L^1$	0	0.2	0.4	0.2	0.4
$L^2$	0.4	0	0.4	0.2	0.4
$L^3$	0.4	0.2	0	0.2	0.4
$L^4$	0.2	0.4	0.2	0	0.2
$L^5$	0.4	0.4	0.4	0.2	0

The set-up costs are given in Table 3. We assume that the weight is given by the user of the algorithm.

Table 3: Set-up costs

$\wedge$	$L^1$	$L^2$	$L^3$	$L^4$	$L^5$
$L^1$	0	5	10	5	10
$L^2$	10	0	10	5	10
$L^3$	10	5	0	5	10
$L^4$	5	10	5	0	5
$L^5$	10	10	10	5	0



We set  $T_m = 3$ .

The inventory and backloging costs are given in Table 4. The inventory cost is the cost of keeping in stock one unit of product during one elementary period while the backloging cost is the cost incurred when one unit of product is missing during one elementary period.

Table 4: Inventory and backloging costs

Product types	Inventory cost	Backloging cost
P <sub>1</sub>	3	42
P <sub>2</sub>	4	35

In Table 5, we provide the solutions for different values of  $w$ . These solutions are obtained in less than 10 seconds CPU.

Table 5: Near-optimal solutions

Weight $w$	Near-optimal cost	Near-optimal solution
0	1732.6	$\langle 2L^0, 5L^2, L^0, 3L^1, 5L^0, 8L^4, 16L^0, 8L^4, 6L^0 \rangle$
1	1747.6	$\langle 2L^0, 5L^2, L^0, 3L^1, 5L^0, 8L^4, 16L^0, 8L^4, 6L^0 \rangle$
2	1762.6	$\langle 2L^0, 5L^2, L^0, 3L^1, 5L^0, 8L^4, 16L^0, 8L^4, 6L^0 \rangle$
5	1866.8	$\langle 2L^0, 5L^2, L^0, 3L^1, 4L^0, 8L^4, 17L^0, 8L^4, 6L^0 \rangle$
10	2039.8	$\langle 2L^0, 5L^2, L^0, 9L^2, 17L^0, 5L^2, 12L^0, 2L^5 \rangle$

### 3.4. The optimal solution

The optimal solution can be reached using a branch-and-bound (B&B) approach. All the parameters previously introduced are used in this approach.

The upper bound required in such an approach is obtained by applying HSCHEDULE. Note that this upper bound can be improved by applying HSCHEDULE at some nodes of the B&B approach for the remaining period.

The lower bound to be computed at each node of the B&B is obtained by completing the partial schedule already obtained by a set of lots for which the set-up times are neglected or replaced by the smallest set-up time involved, and which can be only partially manufactured.

Assume, for instance, that the partial schedule obtained at a given node is:

$$\tilde{L}(K) = (L^{i_1}, L^{i_1}, \dots, L^{i_1}, L^{i_2}, L^{i_2}, \dots, L^{i_2}, \dots, L^{i_k}, L^{i_k}, \dots, L^{i_k})$$

and that the completion time of  $\tilde{L}(K)$  is  $\lambda_K$ , with  $h\Delta \leq \lambda_K < (h+1)\Delta$ .

We denote by  $C_K$  the partial cost associated with  $\tilde{L}(K)$  and by  $S_K$  the inventory vector at time  $\lambda_K$ .

We denote by  $x_q^j$  the number of lots of type  $L^q$  manufactured during period  $[j\Delta, (j+1)\Delta)$  if  $j > h$  or period  $[\lambda_K, (h+1)\Delta)$  if  $j = h$ . The  $x_q^j$  values are real positive values. In other words, we assume that a lot can be partially manufactured and, if so, that the ratios of the products in these partial lots are those of the complete lot.

With these hypotheses, the cost to be paid on  $[\lambda_K, (h+1)\Delta)$  is:

$$C_h = ([S_K]^+ I^t + [-S_K]^+ B^t)((h+1)\Delta - \lambda_K)$$

Similarly, the cost to be paid on  $[(h+1)\Delta, (h+2)\Delta)$  is:

$$C_{h+1} = \left( \left[ S_K + \sum_{q=1}^Q x_q^h L^q - D^{h+1} \right]^+ I^t + \left[ D^{h+1} - S_K - \sum_{q=1}^Q x_q^h L^q \right]^+ B^t \right) \Delta$$

Broadly, we assume that the cost to be paid on period  $[r\Delta, (r+1)\Delta)$  is:

$$C_r = \left( \left[ S_K + \sum_{q=1}^Q \left( \sum_{j=h}^{r-1} x_q^j \right) L^q - \sigma_{h+1}^r \right]^+ I^t + \left[ \sigma_{h+1}^r - S_K - \sum_{q=1}^Q \left( \sum_{j=h}^{r-1} x_q^j \right) L^q \right]^+ B^t \right) \Delta \quad (4)$$

with  $\sigma_{h+1}^r = \sum_{s=h+1}^r D_s$ .

Relation (4) holds for  $r = h+1, \dots, H-1$ .

A lower bound of the cost on  $[\lambda_K, H\Delta]$  is the minimal value of:

$$C_m = \sum_{r=h}^{H-1} C_r + C_K = ([S_K]^+ I^t + [-S_K]^+ B^t)((h+1)\Delta - \lambda_K) + \Delta \sum_{r=h+1}^{H-1} \left( \left[ S_K + \sum_{q=1}^Q \left( \sum_{j=h}^{r-1} x_q^j \right) L^q - \sigma_{h+1}^r \right]^+ I^t + \left[ \sigma_{h+1}^r - S_K - \sum_{q=1}^Q \left( \sum_{j=h}^{r-1} x_q^j \right) L^q \right]^+ B^t \right) + C_K \quad (5)$$

under the constraints:

$$\sum_{q=1}^Q \left( \left[ \sum_{j=h}^{H-1} x_q^j \right] \bar{\theta}^q \right) < H\Delta - \lambda_K \quad (6)$$

and:

$$x_q^j \geq 0 \text{ for } q = 1, \dots, Q \text{ and } j = h, h+1, \dots, H-1 \quad (7)$$

The problem to be solved can be written as the following LP problem, denoted by  $P(\lambda_K, S_K, H)$ :

$$\text{Minimise } \sum_{r=h+1}^{H-1} (Y_r I^t + Z_r B^t)$$

s.t.

$$Y_r \geq S_K + \sum_{q=1}^Q \left( \sum_{j=h}^{r-1} x_q^j \right) L^q - \sigma_{h+1}^r$$

$$Z_r \geq \sigma_{h+1}^r - S_K - \sum_{q=1}^Q \left( \sum_{j=h}^{r-1} x_q^j \right) L^q$$

$$Y_r \geq 0 \text{ and } Z_r \geq 0$$

for  $r = h+1, h+2, \dots, H-1$

Constraints (6) and (7) also hold.

Remember that  $h = \lfloor \lambda_K / \Delta \rfloor$ .

Finally:

$$C_m = C_K + ([S_K]^+ I^t + [-S_K]^+ B^t)((h+1)\Delta - \lambda_K) + \Delta C^*$$

where  $C^*$  is the optimal value of the criterion of problem  $P(\lambda_K, S_K, H)$ .

The B&B algorithm can be presented as follows, assuming that the levels of the B&B approach are numbered starting from 0 (root level). We assume that the data quoted in point 1 of algorithm HSCHEDULE are available.

### B&B algorithm

0. Compute the upper bound using HSCHEDULE. Let  $CC^*$  be this upper bound.

1. ROOT: Set  $\tau_0^1 = 0$ ,  $\tilde{L}_0^1 = \emptyset$ ,  $L_0^1 = L^{i_0}$ ,  $C_0^1 = 0$

*The root is at level 0 and is the only node at this level. In the notations, the low index refers to the B&B level while the top index is the rank of the node at this level.*

2. Set  $m_0 = 1$ .

*$m_0$  is the number of nodes at the current level.*

3. Set  $T = 1$

*$T$  is the level of the B&B algorithm.*

4. Set  $m = 0$

*$m$  is the counter of the number of nodes at level  $T$ .*

5. Set  $cm = +\infty$  and  $mm = 0$

*cm (resp. mm) will contain the minimal average cost (resp. the rank of the node corresponding to the minimal average cost) at level T. These parameters will be used to refresh the upper bound.*

6. For  $i = 1$  to  $m_0$ , do

6.1. Set  $\tau = \tau_{T-1}^i$ ,  $\tilde{L} = \tilde{L}_{T-1}^i$ ,  $C = C_{T-1}^i$ ,  $L^{i0} = L_{T-1}^i$

*$\tau_{T-1}^i$  is the completion time of the last elementary sequence of the  $i$ -th node of the  $(T-1)$ -th level of the B&B.*

*$\tilde{L}_{T-1}^i$  is the whole sequence obtained for the  $i$ -th node of the  $(T-1)$ -th level of the B&B.*

*$C_{T-1}^i$  is the cost associated with  $\tilde{L}_{T-1}^i$ .*

*$L_{T-1}^i$  is the last type of lot produced at the  $i$ -th node of the  $(T-1)$ -th level of the B&B.*

6.2. Compute

$$a = \text{Min}(H\Delta - \tau, T_m\Delta)$$

*$a$  denotes the period to be covered by the next elementary sequence when starting from the  $i$ -th node of level  $T-1$ .*

6.3. For  $q = 0$  to  $Q$ , do

6.3.1. Compute  $b = a - \mu(i_0, q)$

6.3.2. If  $b \leq 0$ , then go to 6.4

$$6.3.3. \quad nn = \lceil b / \bar{\theta}^q \rceil$$

*$nn$  is the number of lots of type  $L^q$  in the next elementary sequence.*

6.3.4. Set  $m = m + 1$

$$6.3.5. \quad \text{Set } \tau_T^m = \tau + \mu(i_0, q) + nn\bar{\theta}^q$$

*$\tau_T^m$  is the completion time of the partial sequence corresponding to the  $m$ -th node at level  $T$ .*

$$6.3.6. \quad \text{Set } \tilde{L}_T^m = \tilde{L} \circ \underbrace{\{L^q \circ L^q \circ \dots \circ L^q\}}_{nn \text{ lots}}$$

$$6.3.7. \quad \text{Set } C_T^m = C(S^0, \tilde{L}_T^m) \quad (\text{see relation (3)})$$

*Computation of the lower bound:*

$$6.3.8. \quad \text{Set } h = \lfloor \tau_T^m / \Delta \rfloor$$

6.3.9. Compute

$$S_T^m = S^0 + \bar{U}\tau_T^m - \bar{D}\tau_T^m$$

6.3.10. Solve  $P(\tau_T^m, S_T^m, H)$ . Let  $C^*$  be the optimal value of the criterion.

6.3.11. Compute

$$\mathcal{C} = C_T^m + \left( [S_T^m]^+ I^t + [-S_T^m]^+ B^t \right) ((h+1)\Delta - \tau_T^m) + C * \Delta$$

$\mathcal{C}$  is the lower bound.

6.3.12. If  $\mathcal{C} \geq CC^*$ , then do  $m = m-1$  and go to 6.4

6.3.13. If  $C_T^m / \tau_T^m \geq cm$ , then go to 6.4

6.3.14. Set

$$cm = C_T^m / \tau_T^m$$

$$mm = m$$

$$S_m = S_T^m$$

$$\lambda_m = \lambda_T^m$$

6.4. End of loop q

7. End of loop i

*End of the B&B process.*

8. If  $m = 0$ , then

8.1. Set  $x = +\infty$

8.2. For  $i = 1$  to  $m_0$ , do

*Selection of the best node at level  $T-1$ :*

If  $C_{T-1}^i < x$ , then

8.2.1. Set  $x = C_{T-1}^i$

8.2.2. Set  $\tilde{L} = L_{T-1}^i$

8.3. End of loop i

8.4. Solution is  $\tilde{L}$  and the corresponding cost is  $x$

8.5. End of the B&B process

*Refreshing the upper bound:*

9. Apply algorithm HSCHEDULE with  $\tau_0^1 = \tau_T^m, S_0 = S_T^m$

10. Compute  $C_p = cm \cdot \lambda_m + CC$  where  $CC$  is the value of the criterion obtained by applying HSCHEDULE on  $(\tau_T^m, H\Delta)$  with  $S_T^m$  as initial inventory

11. If  $C_p < CC^*$ , then  $CC^* = C_p$

*In this case, the upper bound is reduced. Note that we try to reduce the upper bound at each level of the B&B process by completing the node corresponding to the minimal average cost at this level of the B&B: this node is likely the one which may led to the smallest upper bound.*

12. Set  $m_0 = m$

13. Go to 4

Indeed, we keep the optimal result only until time  $HA-2T_m$  to make a comparison with the heuristic algorithm.

### A numerical example

We consider the example presented in Section 4.2. The optimal solutions are presented in Table 6. The computation time required to obtain the optimal solution should be compared with the time required to run the heuristic algorithm (less than 10 seconds).

Table 6: Optimal solutions

Weight w	Optimal cost	Optimal solution	CPU time	Difference between optimal and near-opt. cost
0	1673.4	$\langle 2L^0, 5L^2, L^0, 3L^1, 6L^0, 8L^4, 15L^0, 8L^4, 6L^0 \rangle$	34'29"	3.54%
1	1688.4	$\langle 2L^0, 5L^2, L^0, 3L^1, 6L^0, 8L^4, 15L^0, 8L^4, 6L^0 \rangle$	39'11"	3.51%
2	1703.4	$\langle 2L^0, 5L^2, L^0, 3L^1, 6L^0, 8L^4, 15L^0, 8L^4, 6L^0 \rangle$	37'04"	3.48%
5	1748.4	$\langle 2L^0, 5L^2, L^0, 3L^1, 6L^0, 8L^4, 15L^0, 8L^4, 6L^0 \rangle$	40'31"	6.77%
10	1823.4	$\langle 2L^0, 5L^2, L^0, 3L^1, 6L^0, 8L^4, 15L^0, 8L^4, 6L^0 \rangle$	54'39"	11.87%

## 5. CONCLUSION

The approach presented in this paper considers the most difficult problem to be solved when using a hierarchical production management system, that is the definition of the manufacturing times when considering an aggregate level of the hierarchy, i.e. a level where entities are lots (instead of parts) and manufacturing subsystems (instead of machines).

Parameter  $T_m$ , which is the minimal period on which the same type of lots is produced, play a pivotal role for the greater the value of  $T_m$ , the more accurate the manufacturing times of the lots, but the less flexible the system, which may lead to a worst value of the criterion. Thus,  $T_m$  should be considered as a control parameter which makes it possible to disconnect an aggregate level from the next low level. This parameter also allows the user to introduce a part of his expertise by adjusting the value of  $T_m$  according to the state of the whole manufacturing system and, in particular, to the level of work-in-process.

Note that it is also possible to increase the manufacturing times introduced in this paper (by 5 to 10% for instance). In this case, we can reduce the value of  $T_m$ . Further studies will be developed to link the increase of manufacturing times to  $T_m$ .

## BIBLIOGRAPHY

- [1] Bensoussan A., Crouhy M. and Proth J-M., *Mathematical Theory of Production Planning*, Advanced Series in Management, North Holland Publishing, 1983.
- [2] Canuto E., Chritodoulou M., Chu C., Donati F., Gaganis V., Janusz B., Proth J-M., Reithofer W. and Vallauri M., "The ESPRIT Basic Research HIMAC: Hierarchical Management and Control in Manufacturing Systems", 1st IFAC Workshop on Manufacturing Systems: Manufacturing, Modelling, Management and Control, MIM'97, Vienna (Austria), February 1997.
- [3] Canuto E., Donati F. and Vallauri M., "Advances in Manufacturing Algebra: Discrete-Event Dynamic Models of Production Processes", 1st IFAC Workshop on Manufacturing Systems: Manufacturing, Modelling, Management and Control, MIM'97, Vienna (Austria), February 1997.
- [4] Elmaghraby S.G., "The Economic Lot Scheduling Problem (ELSP): Review and Extension", *Management Science*, 24(6), 1978.
- [5] Fox B. and Kenneth A., "MRPII Providing a Natural Hub for Computer Integrated Manufacturing Systems", *Industrial Engineering*, 16(10), 1984.
- [6] Hillion H.P. and Proth J-M., *Mathematical Tools in Production Management*, Competitive Methods in Operations Research and Data Analysis, Plenum Press, 1990.
- [7] Kao P.C.E., "A Multi-Product Dynamic Lot-Size Model with Individual Inventory Costs and Joint Set-Up Costs", *Operations Research*, 27(2), 1979.
- [8] Menipaz E., *Essentials of Production and Operation Management*, Prentice-Hall, Englewood Cliffs, N.J., 1984.
- [9] Orlicky J., *Material Requirements Planning: The New Way of Life in Production and Inventory Management*, McGraw-Hill, New York, 1975.
- [10] Proth J-M. and Xie X.L., "A Demand-To-Manufacturing-Requirement Model based on Predefined Configurations", *European Journal of Operational Research*, 53(3), 1991.
- [11] Wagner H.M. and Whitin T.M., "Dynamic Version of the Economic Lot Size Model", *Management Science*, 5, 1958.
- [12] Zangwill W.I., "A Backlogging Model and a Multi-Echelon Model of a Dynamic Economic Lot Size Production System. A Network Approach", *Management Science*, 15(9), 1969.



---

Unité de recherche INRIA Lorraine  
Technopôle de Nancy-Brabois - Campus scientifique  
615, rue du Jardin Botanique - B.P. 101 - 54602 Villers lès Nancy Cedex (France)

Unité de recherche INRIA Rennes - IRISA, Campus universitaire de Beaulieu 35042 Rennes Cedex (France)  
Unité de recherche INRIA Rhône-Alpes - 46, avenue Félix Viallet - 38031 Grenoble Cedex 1 (France)  
Unité de recherche INRIA Rocquencourt - Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)  
Unité de recherche INRIA Sophia Antipolis - 2004, route des Lucioles - B.P. 93 - 06902 Sophia Antipolis Cedex (France)

---

Éditeur  
INRIA - Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)

ISSN 0249 - 6399



★ R R - 3 1 1 3 ★